IN THE CLAIMS

1. (Previously amended) A method in a signal processor for filtering samples in a digital signal, the method comprising:

generating an approximate filtered sample as a function of two samples of the digital signal wherein the two samples are a first fixed-point sample, A, and a second fixed-point sample, B, wherein generating the approximate filtered sample includes calculating (A+B+1)>>1 wherein the ">>" represents a right-shift;

generating a correction as a function of the two samples; and

generating a filtered sample by modifying the approximate filtered sample with the correction.

2-3. (Cancelled)

4. (Original and withdrawn) The method of claim 3, wherein the signal processor is a microprocessor having an instruction for calculating the function (X+Y+1)>>1, and wherein calculating A+B+1>>1 is performed using the instruction.

5. (Original and withdrawn) The method of claim 4, wherein the microprocessor is an Intel™ microprocessor with SSE or SSE2, and wherein the instruction is the PAVGB instruction.

6. (Original and withdrawn) The method of claim 4, wherein the microprocessor is an Intel™ microprocessor with SSE or SSE2, and wherein the instruction is the PAVGW instruction.

7. (Original and withdrawn) The method of claim 3, wherein generating the approximate filtered sample further includes calculating (A + ((A+B+1)>>1) + 1) >>1.

8. (Previously amended) The method of claim 1, wherein generating the approximate filtered sample further includes:

calculating E = ((A+B+1)>>1)<<S wherein the "<<" represents a left-shift;

calculating F = ((A+B+1)>>1)<<R; and

calculating the approximate filtered sample as E + F;

wherein S and R are positive fixed-point number.

9. (Previously amended) The method of claim 8, wherein generating the correction includes:

calculating Q =~(A ⊕ B) wherein the "~" represents a bit-wise complement;

masking Q with the number one;

calculating G = Q<<(S-1);

calculating H = Q<<(R-1); and

calculating the correction as G + H.

10. (Original) The method of claim 9, wherein generating the filtered sample includes:

calculating the filtered sample as the approximate filtered sampled added with the correction; and

right-shifting the filtered sample by N-1 bits, wherein N is a positive fixed-point number.

11. (Previously amended) The method of claim 1, wherein generating the approximate filtered sample further includes:

calculating E = ((A+B+1)>>1)>>(N-1-S);

calculating F = ((A+B+1)>>1)>>(N-1-R); and

adding E with F;

wherein N, S and R are positive fixed-point numbers, and wherein $N \geq S > R$.

12. (Previously amended) The method of claim 11, wherein generating the correction includes:

calculating Q =~(A ⊕ B) wherein the "~" represents a bit-wise complement;

masking Q with the number one;

calculating G = Q>>(N-S);

calculating H=Q>>(N-R); and

calculating the correction as G + H.

13. (Original) The method of claim 12, wherein generating the filtered sample includes calculating the filtered sample as the approximate filtered sample added with the correction.

14. (Previously amended) The method of claim 1, wherein the two samples are fixed-point numbers, and wherein generating the correction includes:

calculating the correction as the exclusive OR (XOR) of the two samples; and

masking the correction with the integer one.

15. (Original) The method of claim 14, wherein generating the correction further includes, prior to the masking step, generating a bit-wise complement of the correction.

16. (Previously amended) A method in a signal processor for filtering samples in a digital signal, the method comprising:

generating an approximate filtered sample as a function of two samples of the digital signal wherein the two samples are a first fixed-point sample, A, and a second fixed-point sample, B;

generating a correction as $(A \oplus B)$ OR $(A \oplus (A+B>>1))$ wherein the ">>" represents a right-shift;

masking the correction with the number one; and

generating a filtered sample by modifying the approximate filtered sample with the correction.

17. (Presently amended) A method in a signal processor for filtering samples in a digital signal, the method comprising:

generating an approximate filtered sample as a function of two samples of the digital signal wherein the two samples are a first fixed-point sample, A, and a second fixed-point sample, B;

generating a correction as $(A \oplus (A+B>>1))$; wherein the ">>" represents a right-shift;

bit-wise ANDing the correction with the number one; and

generating a filtered sample by modifying the approximate filtered sample with the correction.

18. (Original) The method of claim 1, wherein generating the filtered sample includes adding the correction to the approximate filtered sample.

19. (Previously amended) The method of claim 1, wherein generating the filtered sample includes subtracting the correction from the approximate filtered sample.

20. (Previously amended) The method of claim 1, wherein the microprocessor is an Intel™ microprocessor with MMX™/SSE, wherein the samples A and B are 8 bits wherein the generating the approximate filtered sample, generating the correction and generating the filtered sample include executing the instructions:

```
pxor    C_REG, A_REG;
pand    C_REG, CONST;
pavgb   A_REG, B_REG;
psubb   A_REG, C_REG;
```

wherein A_REG is a register that initially includes the sample A, B_REG is a register that includes the sample B, C_REG is a register that initially includes the sample B, and CONST is a constant that includes the eight-bit number 0x01.

21. (Previously amended) The method of claim 1, wherein the microprocessor is an Intel™ microprocessor with MMX™/SSE, wherein the samples A and B are eight bits wherein the generating the approximate filtered sample, generating the correction and generating the filtered sample include executing the instructions:

```
pxor    C_REG, A_REG;
pandn   C_REG, CONST;
pavgb   A_REG, B_REG;
paddb   A_REG, C_REG;
```

wherein A_REG is a register that initially includes the sample A, B_REG is a register that includes the sample B, C_REG is a register that initially includes the sample B, and CONST is a constant that includes the eight-bit number 0x01.

22. (Original and withdrawn) The method of claim 1, wherein the less than four samples are only three samples.

23. (Original and withdrawn) The method of claim 22, wherein the two samples are a first fixed-point number, A, a second fixed-point number, B, and a third fixed-point number, C, wherein generating the approximate filtered sample includes:

calculating M = (A+B+1)>>1;

calculating $N = (C+0+1)>>1$; and

calculating the approximate filtered sample as $X = (M+N+1)>>1$.

24. (Original and withdrawn) The method of claim 23, wherein generating the correction includes:

calculating the correction as $((A \oplus B) \text{ AND } (C)) \text{ OR } (M \oplus N)$; and

masking the correction with the number one.

25. (Original and withdrawn) The method of claim 23, wherein generating the correction includes:

calculating the correction as $(A \oplus B) \text{ OR } C \text{ OR } (M \oplus N)$; and

masking the correction with the number one.

26. (Original and withdrawn) The method of claim 22, wherein the two samples are a first fixed-point number, A, a second fixed-point number, B, and a third fixed-point number, C, wherein generating the approximate filtered sample includes:

calculating $M = (B+C+1)>>1$; and

calculating the approximate filtered sample as $X = (A+M+1)>>1$.

27. (Original and withdrawn) The method of claim 26, wherein generating the correction includes:

calculating the correction as $A \oplus M$; and

masking the correction with thc number one.

28. (Original and withdrawn) The method of claim 26, wherein generating the correction includes:

calculating the correction as $(B \oplus C)$ OR $(A \oplus M)$;

masking the correction with the number one.

29-42. (Cancelled)

43. (Previously amended) A computer program product comprising:

a computer readable storage medium having computer program code embodied therein for quantizing a digital signal, the computer program code comprising:

code for generating an approximate filtered sample as a function of two samples of the digital signal wherein thc two samples are a first fixed-point sample, A, and a second fixed-point sample, B, wherein generating the approximate filtered sample includes calculating (A+B+1)>>1 wherein the ">>" represents a right-shift;

code for generating a correction as a function of the two samples; and

code for generating a filtered sample by modifying the approximate filtered sample with the correction.

44. (Previously amended) A system for filtering samples in a digital signal, the system comprising:

a memory that stores samples in the digital signal; and

a processor coupled to the memory and operable to perform the method of:

generating an approximate filtered sample as a function of two samples of the digital signal wherein the two samples are a first fixed-point sample, A, and a second fixed-point sample, B, wherein generating the approximate filtered sample includes calculating (A+B+1)>>1 wherein the ">>1" represents a right-shift by one bit;

generating a correction as a function of the two samples; and

generating a filtered sample by modifying the approximate filtered sample with the correction.

45-46. (Cancelled)

App. Ser. No.: 10/057,694
Atty. Doc. No.: D02750

INTERVIEW SUMMARY

On or around June 14, 2006, Applicant's representative spoke with the Examiner regarding the finality of the rejection mailed on June 2, 2006. During this conversation, the Examiner agreed that the finality of that Office Action was premature. Therefore, the June 2, 2006 Office Action has been changed to non-final.